



Licenciatura Engenharia Informática e Multimédia
Instituto Superior de Engenharia de Lisboa
Ano letivo 2022/2023

Modelação e Programação
Relatório: Trabalho Prático 4 - Parte A

Turma: 22D

Nome: Miguel Alcobia

Número: 50746

Professor: João Ventura

Data: 22 de Maio de 2023

Índice

Introdução	1
Objetivos	1
Explicação das classes	2
Exemplo de funcionamento	3

Introdução

Neste trabalho, os alunos devem demonstrar que aprenderam e dominam a matéria lecionada durante o semestre na unidade curricular de Modelação e Programação, ao realizar um projeto que pode ir desde uma aplicação de gestão até um jogo.

Primeiramente os alunos deverão criar a aplicação sem uma interface gráfica, sendo esta o foco da segunda parte do TP4.

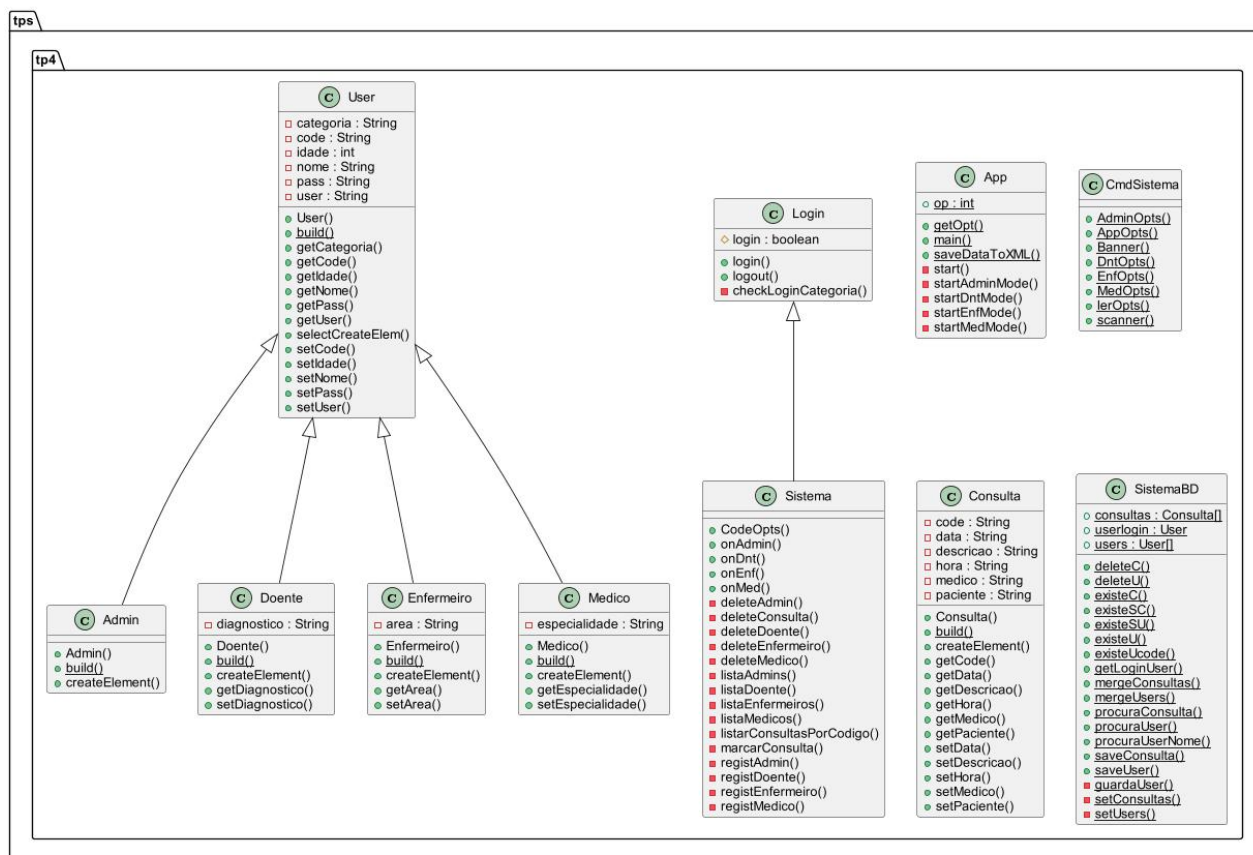
Objetivos

Tomei como objetivo desenvolver uma aplicação para gestão de um hospital. Onde os utilizadores seriam divididos em:

- Administradores
- Médicos
- Enfermeiros
- Doentes

A aplicação começara com um login e depois terá vários menus de acesso, consoante a categoria do utilizador que efetuou o login. O Administrador terá mais opções e terá acesso a dados que outros não terão. Além de registar, apagar, listar estes utilizadores, será possível marcar, registar e apagar consultas. Consultas estas que terão como parâmetros os utilizadores referidos acima, além de dados próprios, tal como data, hora e descrição.

Esta aplicação teria o seguinte UML: (próxima página)



PlantUML diagram generated by Sketchit! (<https://bitbucket.org/pmeasneur/sketchit/>)
For more information about this tool, please contact philippe.measneur@gmail.com

Imagem 1 - UML do programa

Explicação das classes

User - Classe que representa os users e da qual vão estender as classes que representam o Médico, Enfermeiro e Doente e Admin. Não foi criada como abstrata, pois no decorrer no código são criados objetos do tipo User. Nesta classe existe um método selectCreateElem, que consoante a categoria do user, chama o createElement correspondente.

Admin - Classe que representa os admins e que estende do User. No seu construtor já está definida a sua categoria. Nesta classe existe o método createElement para criar um elemento no XML, representando o Admin e um build que constrói um Admin a partir dos dados do XML.

Médico - Classe que representa os médicos e que estende do User. No seu construtor já está definida a sua categoria e tem como atributo próprio a especialidade do médico. Nesta classe existe o método createElement para criar um elemento no XML, representando o Médico e um build que constrói um Médico a partir dos dados do XML.

Enfermeiro - Classe que representa os enfermeiros e que estende do User. No seu construtor já está definida a sua categoria e tem como atributo próprio a área onde o enfermeiro atua. Nesta classe existe o método createElement para criar um elemento no XML, representando o Enfermeiro e um build que constrói um Enfermeiro a partir dos dados do XML.

Doente - Classe que representa os doentes e que estende do User. No seu construtor já está definida a sua categoria e tem como atributo próprio o diagnóstico. Nesta classe existe o método createElement para criar um elemento no XML, representando o Doente e um build que constrói um Doente a partir dos dados do XML.

Login - Classe onde tem toda a lógica para se fazer login e se proceder à sua respetiva validação. Também tem o método para logout.

Sistema - Classe que estende do Login para que o sistema só funcione se o login for bem sucedido. Nesta classe estão os modos de operação para cada categoria de user, pois dependendo dela têm-se acesso a diferentes opções. Além disso é aqui que estão os métodos para os menus de registar, apagar e listar os diferentes users e consultas.

Consulta - Classe que representa uma consulta, com todos os devidos dados no construtor. Nesta classe existe o método createElement para criar um elemento no XML, representando a Consulta e um build que constrói uma Consulta a partir dos dados do XML.

CmdSistema - Classe onde estão os prints dos vários menus que serão impressos na consola. Além disso, existe um método Scanner para realizar scanners e otimizar tempo durante a realização do código; por fim, um método lerOpts que lê a opção escolhida pelo user.

SistemaBD - Classe que simula uma base de dados, já com alguma informação de base e com métodos para funcionamento da base dados como: apagar, procurar, verificar se existem Users e Consultas. Além destes, ainda existem métodos merge para juntar a informação já existente com a do XML e outros de “save” para atualizar dados da base.

App - Método principal da aplicação, onde carrega-se os dados do XML para o sistema antes de iniciar a aplicação em si e apresentar o login o utilizador.

Exemplo de funcionamento

Para exemplificar o funcionamento da app, vamos assumir o contexto de sermos um médico e vamos marcar uma consulta para um paciente.

Vamos assumir o papel da doutora Chiara Petrillo e marcar uma consulta para o Carlos Ferreira:
Os dados são:

```
<medico>
  <nome>Chiara Petrillo</nome>
  <idade>28</idade>
  <code>M13612</code>
  <user>cpetrillo</user>
  <pass>tMtVvQY8BP</pass>
  <especialidade>Nutrição</especialidade>
  <categoria>medico</categoria>
</medico>
```

Procedemos então ao login e ao ser bem sucedido, é apresentado o MENU MÉDICO:

```
=====
===== HOSPITAL =====
=====

[1] -> Login Administrador
[2] -> Login Médico
[3] -> Login Enfermeiro
[4] -> Login Doente
-----
Opção desejada: 2
Username : cpetrillo
Password : tMtVvQY8BP
===== Login com Sucesso =====
-----

**** MENU MÉDICO ****

[1] -> Listar Médicos
[2] -> Listar Doentes
[3] -> Listar Enfermeiros
[4] -> Listar Consultas
[5] -> Marcar Consultas
[6] -> Apagar Consultas
[0] -> Logout
-----
Opção desejada:
```

Marcamos então a consulta:

```
-----
Opção desejada: 5
-----

***** MARCAR CONSULTA *****
-----

| Paciente : Carlos Ferreira
-----
| Médico : Chiara Petrillo
-----
| Descrição : Consulta de Rotina
-----
| Data : 23/05/2023
-----
| Hora : 09:00
-----
| Código (CXXXXX): 002827
-----
Consulta registado com sucesso
Selecione o seu código:
```

Vamos verificar se a consulta está realmente marcada:

```
Consulta registado com sucesso
Selecione o seu código: M13612
-----

**** MENU MÉDICO ****

[1] -> Listar Médicos
[2] -> Listar Doentes
[3] -> Listar Enfermeiros
[4] -> Listar Consultas
[5] -> Marcar Consultas
[6] -> Apagar Consultas
[0] -> Logout
-----

Opção desejada: 4
-----

Insira o código para o filtro desejado: M13612
---- CONSULTAS DO PACIENTE/MÉDICO COM O CÓDIGO M13612 -----
-----

| DATA      | HORA      | DESCRIÇÃO      | DOENTE      |
-----
| 1/12/2023  | 14:30     | Exames aos intestinos | Carlos Ferreira
-----
| 23/05/2023 | 09:00     | Consulta de Rotina   | Carlos Ferreira
-----
-----

Selecione o seu código:
```

Como podemos ver, a consulta foi marcada com sucesso. Agora efetuamos o logout:

```
-----
Selecione o seu código: M13612
-----

**** MENU MÉDICO ****

[1] -> Listar Médicos
[2] -> Listar Doentes
[3] -> Listar Enfermeiros
[4] -> Listar Consultas
[5] -> Marcar Consultas
[6] -> Apagar Consultas
[0] -> Logout
-----

Opção desejada: 0
-----

===== Logout com Sucesso =====
Users salvos no XML com sucesso!

Process finished with exit code 0
```